

**IMPORTANT: CUBE\_TESTER\_AI will only be available from 7:00 am GMT to 9:00 pm GMT**

## **CUBE\_TESTER\_AI: Innovative Testing Solutions for Java**

Welcome to the CUBE\_TESTER\_AI web application! This tool is designed to simplify and automate the process of generating JUnit test cases for Java projects, especially for Spring Boot applications. The generator will create structured and robust test cases, saving you time and effort in writing them manually.

### **Table of Contents**

- Overview
- Features
- Prerequisites
- How to Use
  - Step 1: Initial Input
  - Step 2: Branch & Class Selection
  - Step 3: Confirmation & Test Generation
- Notification & Download

### **Overview**

CUBE\_TESTER\_AI streamlines the creation of JUnit test cases for Spring Boot applications. Simply provide some basic information about your project, and the tool will generate fully functional test classes that follow best practices.

### **Features**

- Automatically generates JUnit test cases based on your Spring Boot project's structure.
- Customizable class selection for generating test cases.
- Provides a download link after a few minutes (depends on the size of the class under test) to retrieve the generated tests.

## Prerequisites

Before using this tool, ensure that your Spring Boot project meets the following requirements:

- A valid Spring Boot project hosted in a Git repository.
- Your project must have the `spring-boot-starter-test` dependency defined in its `pom.xml`:

```
• <dependency>

    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-test</artifactId>

    <scope>test</scope>

</dependency>
```

## How to Use Step 1: Initial Input

On the main page of the web application, you will be asked to fill out the following fields:

1. Name – Enter your name (Some random name). (Optional)
2. Email – Enter your email address (Make a temporary anonymous email at

<https://internxt.com/it/temporary-email>). (Optional, used to receive the generated tests via

email)

3. Git URL of the project – Provide the Git URL for the Spring Boot project. Make sure the

project includes the `spring-boot-starter-test` dependency. Click the "Send" button to proceed to the next step.

## **Step 2: Branch & Class Selection**

On the following page, you'll need to select the appropriate options:

1. Branch – Select the branch of the repository where the JUnit tests should be generated.

2. Java Version – Select the version of Java used in the project.

3. Class Name – Enter the name of the class (without the `.java` suffix) for which you want to

generate tests.

After entering this information, click the "Send" button.

## **Step 3: Confirmation & Test Generation**

On the next page, you will be shown the details of the information you provided. Please review these fields carefully:

- JUnit generation requested by
- ProjectURL
- Branch
- JavaVersion

- **ClassName**

Once you've confirmed everything is correct, click the "Start" button to initiate the test generation process.

## Notification & Download

After you start the process, a notification will appear on the page:

"If you didn't provide your email, click on the button OPEN RESULT and just wait there for a few minutes (depending on the class size) until you see a button to download your generated tests as a zip file."

"The generation process has started successfully. You will receive an email with the link to download the project containing the JUnit tests created through this tool. Save the following link to download the generated code: '<http://link>'."

Make sure to save the link provided to download the generated test cases later.

Thank you for using CUBE\_TESTER\_AI ! We hope it simplifies your testing process and enhances your productivity in developing robust Java applications.

## Example -

### CUBE\_TESTER\_AI

Innovative Testing Solutions for Java

Name

Mark

Email

name@example.com

Please fill the following field with the git url of a spring boot application. The spring boot application must have the spring-boot-starter-test specified as dependency. In example:

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-test</artifactId>
  <scope>test</scope>
</dependency>
```

In the next page a branch selection is mandatory and the name of a class

Git url of the project\*

<https://github.com/Astha-Tiwari02/MongodbCRUD>

\* required fields

Send

Enter the link of a public repository and click send

## CUBE\_TESTER\_AI

### Innovative Testing Solutions for Java

JUnit generation requested by: ()

Project URL: <https://github.com/Astha-Tiwari02/MongodbCRUD>

Branch  
main

Java version  
Java 11

Class name (without the .java suffix)  
ResourceNotFoundException

Send

Choose branch, java version and write class name and then click send

## CUBE\_TESTER\_AI

### Innovative Testing Solutions for Java

JUnit generation requested by: ()

Project URL: <https://github.com/Astha-Tiwari02/MongodbCRUD>

Branch: main

Java Version: 11

Class name: ResourceNotFoundException

Start

Verify all the details and click start

## CUBE\_TESTER\_AI

### Innovative Testing Solutions for Java

The generation process has started successfully. You will receive an email with the link to download the project containing the JUnit tests created through this tool. Save the following link to download the generated code: [http://ec2-54-76-66-44.eu-west-1.compute.amazonaws.com:50105/v2/response?gen\\_id=ca37307a-ddab-4b65-b6a3-6f1e1bbbf7](http://ec2-54-76-66-44.eu-west-1.compute.amazonaws.com:50105/v2/response?gen_id=ca37307a-ddab-4b65-b6a3-6f1e1bbbf7)

Open Result

JUnit generation requested by: ()

Project URL: <https://github.com/Astha-Tiwari02/MongodbCRUD>

Branch: main


Java Version: 11

Class name: ResourceNotFoundException

Save the link given there or click open result

# CUBE\_TESTER\_AI

## Innovative Testing Solutions for Java

Project URL:	https://github.com/Astha-Tiwari02/MongodbCRUD
Branch:	main
Java Version:	11
Class name:	ResourceNotFoundException
Requested on	2024-10-03 08:03:24
File name:	app.log (Generated on 2024-10-03 08:03:33) <a href="#">Download</a>
File name:	config.yaml (Generated on 2024-10-03 08:03:24) <a href="#">Download</a>
Waiting for the process to finish... 	

As the process starts you can see a spinner running and you can even see a `app.log` which updates continously with the logs of docker so you can download it and check if there are any errors.

# CUBE\_TESTER\_AI

## Innovative Testing Solutions for Java

Project URL:	https://github.com/Astha-Tiwari02/MongodbCRUD
Branch:	main
Java Version:	11
Class name:	ResourceNotFoundException
Requested on	2024-10-03 08:03:24
File name:	app.log (Generated on 2024-10-03 08:06:51) <a href="#">Download</a>
File name:	junit.zip (Generated on 2024-10-03 08:06:51) <a href="#">Download</a>

Once the process is completed the spinner disappears and you can see `junit.zip` where you can see the generated junit tests and junit logs as well.